2.38 LATERAL AUTOPILOT

No major errors were found in the Lateral Autopilot Functional Element (FE) for ESAMS 2.6.2. The overall code quality is good, with only a few corrections recommended for the internal documentation.

Table 2.38-1 listed below summarizes the desk-checking and software testing verification activities for each subroutine in the Lateral Autopilot FE. The two results columns contain checks if no discrepancies were found. Where discrepancies were found, the desk check results column contains references to discrepancies listed in Table 2.38-4, while the test case results column lists the number of the relevant test case in Table 2.38-6. More detailed information on the results is recorded in these tables.

TABLE 2.38-1. Verification Results Summary.

Design Element	Code Location	Desk Check Result	Test Case ID	Test Case Result
38-1 Compute Autopilot Input Command	PILOT8 143-150	✔	38-3	✔
38-2 Compute Decoder and Servo Output	TRAN1S 63-81 PILOT8 153-164 201-202	✔	38-2, 4, 5, 11	✔
38-3 Compute Input Minus Feedback	PILOT8 179-194 197-198	✔	38-8, 9, 10	✔
38-4 Calculate Output from Integrator	PILOT8 205-220	✔	38-12, 13, 14	✔
38-5 Convert Body Angular Rotation Rate to Correct Units	PILOT8 167-176	✔	38-6, 7	✔
Initialize and Age Integrator Back Values	PILOT8 131-140 223-232	✔	38-4, 11	✔
Determine Proper Autopilot Routine	AUTOP 57-58	✔	38-1	✔

2.38.1 Overview

A lateral autopilot is the means by which an unmanned missile alters its trajectory to intercept its target. The autopilot does this by positioning the control surfaces to produce the desired directional change based on commands from guidance. In addition to steering to the target, the autopilot must maintain stable and controlled flight. A closed-loop feedback system provides the missile orientation information which is used in the control surface position decision process. This feedback information prevents the missile from developing unstable flight while pursuing a maneuvering target. The design of the autopilot control system includes missile physical characteristics and maneuvering requirements. The type and level of feedback used must be balanced carefully to give optimum performance while maintaining stability.

ESAMS 2.6.2 implementation of Lateral Autopilot for our specific missile system is primarily accomplished with two subroutines, PILOT8 and TRAN1S. TRAN1S is used by most of the other autopilot routines as well. The third subroutine, AUTOP, simply directs ESAMS to the appropriate autopilot routine. The three subroutines used for this FE are described in Table 2.38-2.

TABLE 2.38-2. Subroutine Descriptions.

Module Name	Description
AUTOP	Calls proper autopilot routine based on value of IAP flag.
PILOT8	Performs computations which represent the autopilot Laplace block diagram for the missile system of interest with guidance commands and feedback as inputs and control surface positions as outputs.
TRAN1S	Calculates difference equation coefficients for filters using specific missile constants.

2.38.2 Verification Design Elements

Design Elements defined for the lateral autopilot FE are listed in Table 2.38-1; they are fully described in Section 2.38.2 of ASP II. A Design Element is an algorithm that represents a specific component of the FE design. Design Element 38-1 computes the autopilot input command from the guidance command. Design Element 38-2 computes the outputs from the first order filters, decoder and servo, using a recursive equation. Design Element 38-3 subtracts the feedback signal (two separate feedback signals) from the modified input signal. Design Element 38-4 calculates the output from the integrator using a recursive equation and Design Element 38-5 converts body angular rotation rates to the appropriate units.

TABLE 2.38-3. Missile Movement Design Elements.

Subroutine	Design Element	Description
PILOT8	38-1 Compute Autopilot Input Command	Calculate B.
PILOT8 TRAN1S	38-2 Compute Decoder and Servo Output	Calculate C and H.
PILOT8	38-3 Compute Input Minus Feedback	Calculate E and F.
PILOT8	38-4 Calculate Output from Integrator	Calculate J.
PILOT8	38-5 Convert Body Angular Rotation Rate to Correct Units	Calculate D.
PILOT8	Initialize and Age Integrator Back Values	Set previous values in recursive equations (X_k terms) to 0 or current value.
AUTOP	Determine Proper Autopilot Routine	Selects missile specific autopilot routine.

2.38.3 Desk Checking Activities and Results

The code implementing this FE was manually examined using the procedures described in Section 1.1 of this report. No discrepancies were discovered.

Except as noted in Table 2.41-5 below, overall code quality and internal documentation were evaluated as good. Subroutine I/O were found to match the ASP II descriptions.

TABLE 2.38-4. Code Quality and Internal Documentation Results.

Subroutine	Code Quality	Internal Documentation
AUTOP	OK	The list of subroutines called in the header is incomplete.
PILOT8	OK	The PURPOSE refers to the subroutine as PILOTG not PILOT8. The variables GDCMY and GDCMP listed under Common/GUIDAP are not used.

2.38.4 Software Test Cases

All software testing was performed by running the entire ESAMS model in debug mode. For these tests, ESAMS was run using a sample input file for the specific missile of interest.

TABLE 2.38-5. Software Test Cases for Lateral Autopilot.

Test Case ID	Test Case Description
38-1	OBJECTIVE: Check that correct specific missile autopilot routine is called via IF-ELSE statements in AUTOP. PROCEDURE: 1. Run ESAMS. Set a breakpoint in the AUTOP subroutine at line 57 and deposit the following value for IAP. IAP = 8 2. Step to next executable line of code. VERIFY: ESAMS is properly directed to specific missile autopilot routine. RESULT: OK
38-2	OBJECTIVE: Check calculation of coefficient array, C(), in TRAN1S. PROCEDURE: 1. Run ESAMS. Set a breakpoint in the TRAN1S subroutine at line 1 and deposit the following values for G, W and T. G W T 18.6 0.0 0.787 9.23 10.0 7.123 5.76 12.5 0.01 2. Independently calculate the values for C(1), C(2) and C(3) for the values in step 1. 3. For each set of data, step through the subroutine and note the value assigned to C(1), C(2) and C(3) and compare to independently calculated values. VERIFY: ESAMS values match the independently calculated values. RESULT: OK

DRAFT

TABLE 2.38-5. Software Test Cases for Lateral Autopilot.

Test Case ID	**Test Case Description**
38-3	OBJECTIVE: Check calculation of XA1 and XA2 in PILOT8. PROCEDURE: 1. Run ESAMS. Set a breakpoint in the PILOT8 subroutine at line 143 and note values of AC(2) and AC(3) and compare to MSLD.DAT file. 2. Deposit the following values for ATIME, EK1 and EK2. ATIME EK1 EK2 < AC(2) 1.6 -0.07 = AC(2) 0.0 1.03 > AC(2) -0.889 -0.675 3. Independently calculate the values for XA1 and XA2 for the values in step 2. 4. Note the values of XA1 and XA2 at lines 145-146 or 148-149 and compare to independently calculated values. VERIFY: ESAMS values match the independently calculated values. RESULT: OK
38-4	OBJECTIVE: Check calculation of YA1 and YA2 in PILOT8. PROCEDURE: 1. Run ESAMS. Set a breakpoint in the PILOT8 subroutine at line 153 and verify that YA10, YA20, XA10 and XA20 are zero on first pass and note values of CA(1), CA(2) and CA(3). 2. Deposit the following values for XA1 and XA2. <u>XA1</u> <u>XA2</u> -2.783 0.01128 0.9873 -1.7739 3. Using the values of CA(1), CA(2) and CA(3), independently calculate the values of YA1 and YA2 for the values in step 2. 4. Note the values of YA1 and YA2 at lines 153 and 154 and compare to independently calculated values. 5. Repeat step 1, setting the breakpoint to occur after 2000 passes, the model will now generate values for XA1 and XA2. Note the values of YA10, YA20, XA1, XA10, XA2 and XA20. 6. Using the values of CA(1), CA(2) and CA(3), independently calculate the values of YA1 and YA2 for the values in step 5. 7. Note the values of YA1 and YA2 at lines 153 and 154 and compare to independently calculated values. 8. Step through the program once, breaking at line 153 again, verify that YA10, YA20, XA10 and XA20 are equal to the values of YA1, YA2, XA1 and XA2 respectively from step 5. VERIFY: ESAMS values match the independently calculated values, including values for CA(1), CA(2) and CA(3). RESULT: OK

TABLE 2.38-5. Software Test Cases for Lateral Autopilot.

Test Case ID	Test Case Description
38-5	OBJECTIVE: Check limiting of YA1 and YA2 and calculation of APEK1 and APEK2 in PILOT8. PROCEDURE: 1. Run ESAMS. Set a breakpoint in the PILOT8 subroutine at line 157, note the value of AC(7) and deposit the following values for YA1 and YA2. YA1 / YA2 < neg AC(7) / < neg AC(7) 0 < YA1 < AC(7) / 0 < YA2 < AC(7) neg AC(7) < YA1 < 0 / neg AC(7) < YA2 < 0 > AC(7) / > AC(7) 2. Independently calculate the values of APEK1 and APEK2 for the values in step 1, using the value of AC(7). 3. Note the values of APEK1 at line 163 and APEK2 at line 164 (when the values have been limited) and compare to independently calculated values VERIFY: ESAMS values match the independently calculated values. RESULT: OK
38-6	OBJECTIVE: Check calculation of FBO1 and FBO2 in PILOT8. PROCEDURE: 1. Run ESAMS. Set a breakpoint in the PILOT8 subroutine at line 167, note the value of AC(8) and deposit the following values for OMEG(2) and OMEG(3). OMEG(2) / OMEG(3) -0.2387 / 0.7803 0.0 / -0.0575 1.0038 / 0.0 2. Independently calculate the values of FBO1 and FBO2 for the values in step 1 and using the value of AC(8). 3. Note the values of FBO1 at line 167 and FBO2 at line 168 and compare to independently calculated values VERIFY: ESAMS values match the independently calculated values. RESULT: OK
38-7	OBJECTIVE: Check limiting of FBO1 and FBO2 in PILOT8. PROCEDURE: 1. Run ESAMS. Set a breakpoint in the PILOT8 subroutine at line 171, note the value of AC(9) and deposit the following values for FBO1 and FBO2. FBO1 / FBO2 < neg AC(9) / < neg AC(9) 0 < FBO1 < AC(9) / 0 < FBO2 < AC(9) neg AC(9) < FBO1 < 0 / neg AC(9) < FBO2 < 0 > AC(9) / > AC(9) 2. Independently calculate the values of FBO1 and FBO2 for the values in step 1, using the value of AC(9). 3. Note the values of FBO1 at line 172 and FBO2 at line 175 (when the values have been limited) and compare to independently calculated values. VERIFY: ESAMS values match the independently calculated values. RESULT: OK

TABLE 2.38-5. Software Test Cases for Lateral Autopilot.

Test Case ID	Test Case Description
38-8	OBJECTIVE: Check calculation of FI1 and FI2 in PILOT8. PROCEDURE: 1. Run ESAMS. Set a breakpoint in the PILOT8 subroutine at line 179, note the value of AC(10) and deposit the following value for ATIME. ATIME < AC(10) 2. Step to line 184, note the value of AC(11) and deposit the following values for FBO1 and FBO2. FBO1 FBO2 -3.2387 4.7071 0.0 -0.0575 3. Independently calculate the values of FI1 and FI2 for the values in step 2 and using the value of AC(11). 4. Note the values of FI1 at line 184 and FI2 at line 185 and compare to independently calculated values. 5. Repeat step 1 and deposit the following value for ATIME. ATIME > AC(10) 6. Step to line 181 and deposit the following values for FBO1, APEK1, FBO2 and APEK2. FBO1 APEK1 FBO2 APEK2 0.0 0.869 -0.707 0.0 -1.075 3.776 -2.321 0.9471 2.044 3.113 3.011 1.9999 7. Independently calculate the values of FI1 and FI2 for the values in step 6 and using the value of AC(11). 8. Note the values of FI1 at line 181 and FI2 at line 182 and compare to independently calculated values. VERIFY: ESAMS values match the independently calculated values. RESULT: OK
38-9	OBJECTIVE: Check limiting of FI1 and FI2 in PILOT8. PROCEDURE: 1. Run ESAMS. Set a breakpoint in the PILOT8 subroutine at line 189, note the value of AC(12) and deposit the following values for FI1 and FI2. FI1 FI2 < neg AC(12) < neg AC(12) 0 < FI1 < AC(12) 0 < FI2 < AC(12) neg AC(12) < FI1 < 0 neg AC(12) < FI2 < 0 > AC(12) > AC(12) 2. Independently calculate the values of FI1 and FI2 for the values in step 1, using the value of AC(12). 3. Note the values of FI1 at line 190 and FI2 at line 193 (when the values have been limited) and compare to independently calculated values. VERIFY: ESAMS values match the independently calculated values. RESULT: OK

TABLE 2.38-5. Software Test Cases for Lateral Autopilot.

Test Case ID	Test Case Description
38-10	OBJECTIVE: Check calculation of XB1 and XB2 in PILOT8. PROCEDURE: 1. Run ESAMS. Set a breakpoint in the PILOT8 subroutine at line 197 and deposit the following values for FI1, CON1D, FI2 and CON2D. FI1 CON1D FI2 CON2D 0.0 0.869 -0.707 0.0 -1.075 3.776 -2.321 0.9471 2.044 3.113 3.011 1.9999 2. Independently calculate the values of XB1 and XB2 for the values in step 1. 3. Note the values of XB1 at line 197 and XB2 at line 198 and compare to independently calculated values. VERIFY: ESAMS values match the independently calculated values. RESULT: OK
38-11	OBJECTIVE: Check calculation of YB1 and YB2 in PILOT8. PROCEDURE: 1. Run ESAMS. Set a breakpoint in the PILOT8 subroutine at line 201 and verify that YB10, YB20, XB10 and XB20 are zero on first pass and note values of CB(1), CB(2) and CB(3). 2. Deposit the following values for XB1 and XB2. XB1 XB2 -20.783 10.01128 10.9873 -11.7739 3. Using the values of CB(1), CB(2) and CB(3), independently calculate the values of YB1 and YB2 for the values in step 2. 4. Note the values of YB1 and YB2 at lines 201 and 202 and compare to independently calculated values. 5. Repeat step 1, setting the breakpoint to occur after 2000 passes, the model will now generate values for XB1 and XB2. Note the values of YB10, YB20, XB1, XB10, XB2 and XB20. 6. Using the values of CB(1), CB(2) and CB(3), independently calculate the values of YB1 and YB2 for the values in step 5. 7. Note the values of YB1 and YB2 at lines 201 and 202 and compare to independently calculated values. 8. Step through the program once, breaking at line 201 again, verify that YB10, YB20, XB10 and XB20 are equal to the values of YB1, YB2, XB1 and XB2 respectively from step 5. VERIFY: ESAMS values match the independently calculated values, including values for CB(1), CB(2) and CB(3). RESULT: OK

TABLE 2.38-5. Software Test Cases for Lateral Autopilot.

Test Case ID	**Test Case Description**
38-12	OBJECTIVE: Check calculation of YC1 and YC2 in PILOT8. PROCEDURE: 1. Run ESAMS. Set a breakpoint in the PILOT8 subroutine at line 205, note the value of DTA and AC(16), and deposit the following values for YC10, YB1 and YB10. YC10 YB1 YB10 0.0 11.345 -9.667 -11.075 7.549 7.776 2.044 3.113 2.011 5.005 -7.753 -8.099 2. Independently calculate the values of YC1 for the values in step 1. 3. Note the values of YC1 at line 205 and compare to independently calculated values. 4. Repeat step 1 and deposit the following values for YC20, YB2 and YB20. YC20 YB2 YB20 0.0 11.345 -9.667 -11.075 7.549 7.776 2.044 3.113 2.011 5.005 -7.753 -8.099 5. Independently calculate the values of YC2 for the values in step 1. 6. Note the values of YC2 at line 206 and compare to independently calculated values. VERIFY: ESAMS values match the independently calculated values. RESULT: OK

TABLE 2.38-5. Software Test Cases for Lateral Autopilot.

Test Case ID	Test Case Description
38-13	OBJECTIVE: Check limiting of YC1 and YC2 and calculation of CON1D and CON2D in PILOT8. PROCEDURE: 1. Run ESAMS. Set a breakpoint in the PILOT8 subroutine at line 209, note the value of AC(17) and deposit the following values for YC1 and YC2. YC1 — YC2 < neg AC(17) — < neg AC(17) 0 < YC1 < AC(17) — 0 < YC2 < AC(17) neg AC(17) < YC1 < 0 — neg AC(17) < YC2 < 0 > AC(17) — > AC(17) 2. Independently calculate the values of YC1 and YC2 for the values in step 1, using the value of AC(17). 3. Note the values of CON1D at line 215 and CON2D at line 216 and compare to independently calculated values. VERIFY: ESAMS values match the independently calculated values. RESULT: OK
38-14	OBJECTIVE: Check calculation of CON1 and CON2 in PILOT8. PROCEDURE: 1. Run ESAMS. Set a breakpoint in the PILOT8 subroutine at line 219 and deposit the following values for CON1D and CON2D. CON1D — CON2D -33.56 — -12.334 0.0 — 2.321 22.113 — 13.011 2. Independently calculate the values of CON1 and CON2 for the values in step 1. 3. Note the values of CON1 at line 219 and CON2 at line 220 and compare to independently calculated values. VERIFY: ESAMS values match the independently calculated values. RESULT: OK

2.38.5 Conclusions and Recommendations

2.38.5.1 Code Discrepancies

No code discrepancies were found.

2.38.5.2 Code Quality and Internal Documentation

The autopilot Laplace block diagram was properly and effectively represented in the code. Some of the internal documentation was incomplete or incorrect. Additional comments and corrections are recommended.

2.38.5.3 External Documentation

There is no external documentation for ESAMS 2.6.2. Therefore, the external documentation for ESAMS 2.5 was used. Other than choosing which missile to use, there is no direct user interface to lateral autopilot FE, therefore, it is not discussed in the *User's Manual* [3]. The method used to solve the Laplace transforms used in the filters is addressed satisfactorily in the *Analyst Manual* [4].

DRAFT